Configuring Access Using Nomad ACLs

Eric Wright
Technology Evangelist, Podcaster

@DiscoPosse www.discoposse.com www.discopossepodcast.com

Overview

- Learn about how ACLs can limit access to Nomad resources
- Explore the process of implementing ACLs
- Understand the limitations of Nomad ACLs
- Discover Day 1 and Day 2 operational procedures for ACL management
- Map the Wired Brain Coffee ACL use-case
- Bootstrap an ACL environment in our lab

Nomad ACL Primer

The Elements of a Nomad ACL

Tokens

Authentication represented as a X-Nomad-Token in the header

Policies

One or more rules to limit or grant access to capabilities on specific objects

Capabilities

Actions that are allowed/denied by policy rules

Where Can ACLs be Applied?

Namespace

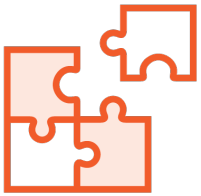
Host Volume

Agent

Node

Where Can ACLs be Applied?

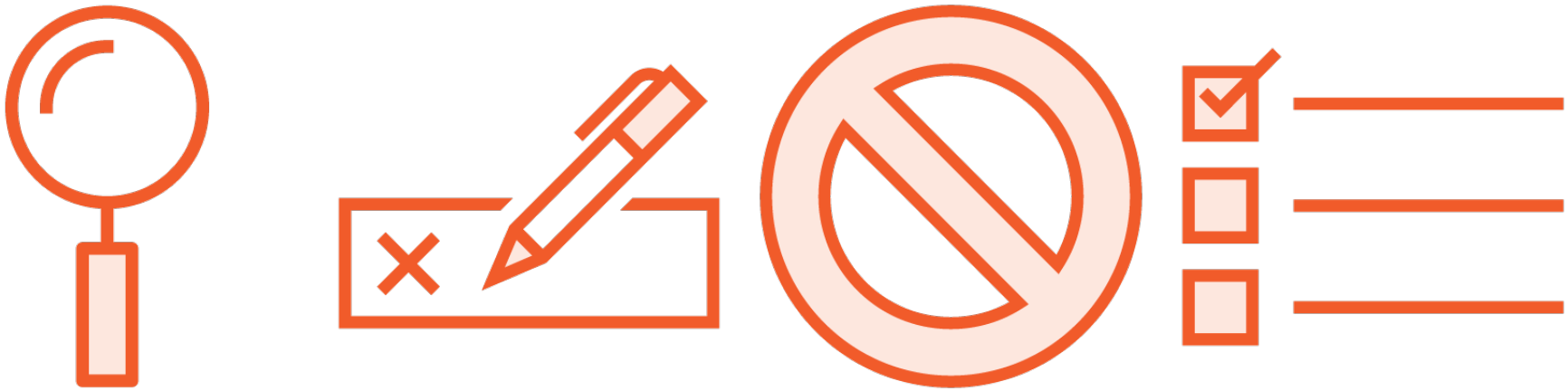
Operator



Plugin

Quota

Permissions for Nomad ACL Policy Rules



Read **Write** **Deny** **List**

Deny

Deny blocks read and modify capabilities, also known as read and write

Deny policies take precedence any other conflicting policy rules to ensure the safety of the environment

As soon as you bootstrap the Nomad ACL subsystem, the default deny policy is enabled unless you have an explicit anonymous policy configured

Nomad ACL Capabilities

Understanding Policy Capabilities

Fine-grained
(list-scaling-policies, scale-job)

Coarse-grained
(read, write)

Read

Namespace Rules

```
list-jobs
read-job
csi-list-volume
csi-read-volume
list-scaling-policies
read-scaling-policy
read-job-scaling
```

Write

Namespace Rules

```
list-jobs
read-job
submit-job
dispatch-job
read-logs
read-fs
alloc-exec
alloc-lifecycle
csi-write-volume
csi-mount-volume
list-scaling-policies
read-scaling-policy
read-job-scaling
scale-job
```

Scale

Namespace Rules

```
list-scaling-policies
read-scaling-policy
read-job-scaling
scale-job
```

Setting an Anonymous ACL Policy

Setting the Anonymous Policy

anonymous.policy

```
namespace "*" {
  policy       = "write"
  capabilities = ["alloc-node-exec"]
}

agent {
  policy = "write"
}

operator {
  policy = "write"
}

quota {
  policy = "write"
}

node {
  policy = "write"
}

host_volume "*" {
  policy = "write"
}
```

Implementing Nomad ACLs

Policy Configuration Options

disco.policy

```
namespace "disco" {
  policy       = "read"
  capabilities = ["submit-job"]
}

namespace "discoposse-*" {
  policy = "read"
}

namespace "discoposse-web" {
  policy = "write"
}
```

Bootstrapping the Nomad ACL Environment

Don’t forget to set your initial policy!

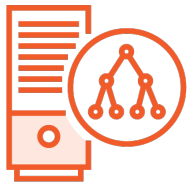
server.hcl (partial)

```
acl {
     enabled = true
}
```

```
$ nomad acl bootstrap
Accessor ID  = 7c7ad453-c3f7-6814-97dc-fcfe6dba6ea5
Secret ID    = 8784ec35-95d4-8258-61c3-0c066d0a45c5
Name         = Bootstrap Token
Type         = management
Global       = true
Policies     = n/a
Create Time  = 2021-06-11 17:38:10.999089612 +0000 UTC
Create Index = 7
Modify Index = 7
```

ACL Limitations

What Are the Limits of Nomad ACLs



Nomad system-specific

Limited set of capabilities

Unable to work at job/net level

For much more granular controls and network-level ACL capabilities you should leverage Consul or a similar system

Day 1 and Day 2 ACL Operations

Day 0, Day 1, and Day 2

Enable anonymous policy, bootstrap ACL subsystem

Create new policies to match new namespaces as they are deployed

Validate and audit policies as part of SOP

Wired Brain Coffee ACL Example

Persona-Based ACLs

Ops Team

Needs broad system-level access for the Nomad infrastructure where devs deploy

Development Team

Needs full access to deploy apps but must have isolation between environments

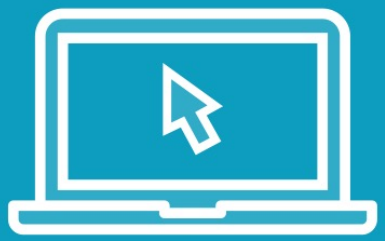
Namespace Policy for Wired Brain Coffee

Operations Team	Development Team
namespace [prod-web-01]	namespace [prod-web-01]
list-jobs	list-jobs
read-job	read-job
-	submit-job
-	dispatch-job
-	read-logs
csi-read-volume	-
csi-list-volume	-

Namespace Policy for Wired Brain Coffee

Operations Team	Development Team
namespace [dev-dallas-dc1]	namespace [dev-dallas-dc1]
list-jobs	list-jobs
read-job	read-job
submit-job	submit-job
dispatch-job	dispatch-job
read-logs	read-logs
csi-read-volume	-
csi-list-volume	-

Demo



- Configure our basic ACL policy
- Bootstrap the ACL environment
- Test our ACL is working as expected

Summary

- Learned about Nomad ACLs
- Explored the implementation process
- Learned the limitations of Nomad ACLs
- Reviewed simple operations
- Mapped the Wired Brain Coffee use-case
- Bootstrapped an ACL environment in our lab

Up Next:
Managing and Monitoring Jobs